Inverted Pendulum control law computation

Applied example from the report on the same subject

Valérian Grégoire--Bégranger - 2024

```
clc; clear; close all;

%% Declaration of the symbols
syms x dx ddx th dth ddth real; % State variables
syms F m M l g real; % Parameters
```

Homogeneous equations of motion (from Wikipedia) :

$(\ddot{x} \quad \ddot{\theta})^T =$

```
% Definition of the equations
eq1 = (F - m*l*dth^2*sin(th) + m*l*ddth*cos(th))/(M + m) - ddx;
eq2 = (ddx*cos(th) + g*sin(th))/(l) - ddth;
system = [eq1;
          eq2];
vars = [ddx ddth];

% Display of the equations
disp(sym(system + [ddx;ddth]));
```

$$\begin{pmatrix} \dfrac{-l\, m \sin(\mathrm{th})\, \mathrm{dth}^2 + F + \mathrm{ddth}\, l\, m \cos(\mathrm{th})}{M + m} \\ \dfrac{\mathrm{ddx} \cos(\mathrm{th}) + g \sin(\mathrm{th})}{l} \end{pmatrix}$$

These equations are coupled, $\ddot{\theta}$ appears in the equation of $\ddot{x}$, and vice-versa. We therefore solve the system of equations for $\ddot{x}$ and $\ddot{\theta}$, decoupling them in the process.

```
%% Solution of the system
```

```
sol = solve(system, vars);
```

Solutions of the system :

```
%% Display of the solution
sols = [sol.ddx sol.ddth];
for i    = 1:2
    disp(string(vars(i)) + " =");
    disp(sym(sols(i)));
    disp(" ");
end
```

ddx =

$$\frac{-l\,m\sin(\mathrm{th})\,\mathrm{dth}^2 + F + g\,m\cos(\mathrm{th})\sin(\mathrm{th})}{-m\cos(\mathrm{th})^2 + M + m}$$

ddth =

$$\frac{-l\,m\cos(\mathrm{th})\sin(\mathrm{th})\,\mathrm{dth}^2 + F\cos(\mathrm{th}) + g\,m\sin(\mathrm{th}) + M\,g\sin(\mathrm{th})}{l\,(-m\cos(\mathrm{th})^2 + M + m)}$$

Computation of the Jacobian matrix :

We can now compute the Jacobian matrix associated with $\ddot{x}$ and $\ddot{\theta}$, with respect to x, $\dot{x}$, θ, and $\dot{\theta}$.

```
vars_ = [x dx th dth];
J = jacobian(sols,vars_)
```

J =

$$\left(\begin{matrix} 0 & 0 & -\frac{l\,m\,\mathrm{dth}^2\cos(\mathrm{th}) - g\,m\cos(\mathrm{th})^2 + g\,m\sin(\mathrm{th})^2}{\sigma_1} - \frac{2\,m\cos(\mathrm{th})\sin(\mathrm{th})\,(-l\,m}{} \\ 0 & 0 & \frac{g\,m\cos(\mathrm{th}) - F\sin(\mathrm{th}) + M\,g\cos(\mathrm{th}) - \mathrm{dth}^2\,l\,m\cos(\mathrm{th})^2 + \mathrm{dth}^2\,l\,m\sin(\mathrm{th})^2}{l\,\sigma_1} - \frac{2\,m\cos(\mathrm{th})\sin(\mathrm{th})\,(-}{} \end{matrix}\right.$$

where

$$\sigma_1 = -m\cos(\mathrm{th})^2 + M + m$$

Selection of a stationary point :

Since we have the Jacobian matrix, we can use a stationary point around which the equations of motion will be linearized. Here, we use the vertical position of the pendulum at any translation in space.

$Stationary\ \ point =$

```
stat_point = [x 0 0 0];
disp(sym(stat_point'));
```

$$\begin{pmatrix} x \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Application of the stationary point to the Jacobian :

By substituting the variables by their values at the stationary point, we get a simplified Jacobian matrix.

$\bar{J} =$

```
J_ = subs(J,vars_,stat_point);
disp(J_);
```

$$\begin{pmatrix} 0 & 0 & \frac{g\,m}{M} & 0 \\ 0 & 0 & \frac{M\,g + g\,m}{M\,l} & 0 \end{pmatrix}$$

Writing the A matrix from the Jacobian :

Using the Jacobian matrix, the A matrix becomes

$A =$

```
A = [0 1 0 0;
     J_(1,:);
     0 0 0 1;
     J_(2,:)];
disp(sym(A));
```

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{g\,m}{M} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{M\,g + g\,m}{M\,l} & 0 \end{pmatrix}$$

Looking at A matrix's eigenvalues :

$Sp\,(A) =$

```
eigs = eig(A);
disp(sym(eigs));
```

$$\begin{pmatrix} 0 \\ 0 \\ -\sqrt{\frac{g\ (M+m)}{M\ l}} \\ \sqrt{\frac{g\ (M+m)}{M\ l}} \end{pmatrix}$$

The system is unstable as there is a positive real-valued eigenvalue. This was expected, but now we have to check if the system is controllable.

Getting the B matrix :

The B matrix describes how the force F applied to the cart will influence the state variables. Solving the coupled equations of motion around the stationary point is a good approach to obtain the influence of F on the system dynamics. We can also normalize the B matrix by dividing it by F to get a simpler representation of the influence of F.

$B =$

```
% Putting stationary points in the initial equations
eqF1 = subs(eq1,[x dx th dth], [x 0 0 0]);
eqF2 = subs(eq2,[x dx th dth], [x 0 0 0]);
sysF = [eqF1, eqF2];
solF = solve(sysF,vars);

% Solving for F in ddx and ddth
B = [0;
     solF.ddx;
     0;
     solF.ddth] / F; % Dividing by F to normalize the scaling of B
disp(sym(B));
```

$$\begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{M\ l} \end{pmatrix}$$

Parameters values :

Overall, the sate-space representation of the system is the following :

$\dot{x} = Ax + Bu$

Let's check if the system is controllable.

Values for the parameters :

$(g \quad m \quad M \quad l)^T =$

```
parameters = [g m M l];
param_vals = [9.81 0.015 0.120 0.06];
disp(sym(param_vals'));
```

$$\begin{pmatrix} \frac{981}{100} \\ \frac{3}{200} \\ \frac{3}{25} \\ \frac{3}{50} \end{pmatrix}$$

Controllability matrix :

Before trying to build a controller, we need to make sure that the system can be controlled at all with the current actuators (the force F along the x axis).

$C =$

```
A_ = double(subs(A,parameters,param_vals));
B_ = double(subs(B,parameters,param_vals));
C = ctrb(A_,B_);
disp(sym(C));
```

$$\begin{pmatrix} 0 & \frac{25}{3} & 0 & \frac{2725}{16} \\ \frac{25}{3} & 0 & \frac{2725}{16} & 0 \\ 0 & \frac{1250}{9} & 0 & \frac{204375}{8} \\ \frac{1250}{9} & 0 & \frac{204375}{8} & 0 \end{pmatrix}$$

Controllability matrix's rank :

The rank of the controllability matrix indicates whether the system is controllable or not.

$Rank\,(C) =$

```
rk = rank(C);
disp(sym(rk));
```

$$4$$

The rank of the controllability matrix C being equal its number of rows/columns means that the system is indeed controllable. Otherwise, one would have needed to add actuators to control the system.

Eigenvalues placement :

The system is therefore controllable, although not stable currently. Let's place some eigenvalues for the system by choosing an appropriate K matrix. As long as all the placed eigenvalues have negative real parts, the system will be returning towards a stabilized state. However, the more negative the spectral abscissa (less negative eigenvalue) is, the faster the stabilization will be. Past some value, the system becomes unstable. Eigenvalues close to 0 are usually more suitable for a realistic system.

$K =$

```
K = place(A_,B_,linspace(-1,-4,4));
disp(K);
```

```
   -0.0176   -0.0367    1.5774    0.0742
```

Placed eigenvalues verification :

We can now verify that the eigenvalues are correctly placed.

$Sp\,(A - BK) =$

```
eigs_ = eig(A_-B_*K);
disp(eigs_);
```

```
   -4.0000
   -3.0000
   -2.0000
   -1.0000
```

The eigenvalues are properly placed. Therefore we have obtained correct A, B, and K matrices for the controller.

These matrices can then be sent to simulink for simulation. To control the system, one must solve the $\dot{x} = Ax + Bu = Ax - K(x - setpoint)$ equation.

Running the simulation :

$Starting\ point =$

```
%% Starting position for the inverted pendulum
starting_position = [20; 0;
                    -15; 0];

%% Display of the vector
disp(sym(starting_position));
```

$$\begin{pmatrix} 20 \\ 0 \\ -15 \\ 0 \end{pmatrix}$$

$Target\ state =$

```
%% Target state for the inverted pendulum
target_position = [0; 0;
                   0; 0];

%% Display of the vector
disp(sym(target_position));
```

$$\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Running the simulation

```
run("simulation.slx")
```